TITLE:

STRUCTURAL ANALYSIS OF ANTOMATA NETWORK : STRUCTURALLY REDUCED NETWORK(Mathematical Foundations of Computer Science and Their Applications)

AUTHOR(S):

Nishio, H.; Saito, T.; Morita, K.

CITATION:

Nishio, H. ...[et al]. STRUCTURAL ANALYSIS OF ANTOMATA NETWORK : STRUCTURALLY REDUCED NETWORK(Mathematical Foundations of Computer Science and Their Applications). 数理解析研究所講究録 1985, 556: 163-170

ISSUE DATE:

1985-04

URL:

http://hdl.handle.net/2433/98962

RIGHT:

京都大学学術情報リポジトリ
KURENAI 紅
Kyoto University Research Information Repository

数理解析研究所講究録
556巻 1985年 163-170

STRUCTURAL ANALYSIS OF AUTOMATA NETWORK
--STRUCTURALLY REDUCED NETWORK--

オートマタ・ネットワークの構造の解析

by H.Nishio,T.Saito and K.Morita
西尾英之助, 斉藤隆, 森田浩一
Department of Biophysics, Kyoto Univ.

#1. Introduction

The automata network was defined to be an edge labeled directed graph with element finite automata allocated at inner vertices(1),(2),(3). As was discussed in the concluding remarks of our previous paper(3), we consider here an automata network as a graph with vertex labeling fundtion $\beta:V\rightarrow H$, where H is the set of colors. Note that the function β specifies a "color" from H to each inner vertex, but does not an element automaton itself. (When we investigate the situation where it is essential what actual element automaton is to be placed at each vertex, we should introduce another one to one function $\gamma:H\rightarrow M$ where M is the set of all finite automata.) The countable set of colors H is taken to be fixed during our whole study.

Thus an automata network, which is under investigation here, will be called a colored network and expressed in the form (G,β) where $G=(V',E)$, $V'=V\cup V_I$ where V_I is the set of input vertices and $E=(E_1,E_2,\ldots,E_k)$ where $E_i\subset V\times V'$ specifies edges which are labeled with i.

<u>Definition 1.</u> Inverted graph

Let $A=(G,\beta)$ be a colored network. The inverted graph $\overline{G}$ of G is defined to be an edge labeled directed graph $(V',\overline{E})$ where $\overline{E}=(\overline{E}_1,\overline{E}_2,\ldots,\overline{E}_k)$, $E_i\subset V'\times V$ and $(u,v)\in\overline{E}_i$ if and only if $(v,u)\in E_i$.

From the definition of E_i (see (1)), it will be seen that in $\overline{G}$ at most one edge having the label i goes out from each inner vertex.

<u>Definition 2</u> Conjugate machine

Let V' be finite and $\overline{A}=(\overline{G},\beta)$. Then $\overline{A}$ can be considered to be an incompletely specified sequential Moore machine where the state set is V', the input alphabet is $K=\{1,2,\ldots,k\}$, the state transition is determined by $\overline{E}$ and the output function is β. If $(u,v)\in\overline{E}_i$ (that is $(v,u)\in E_i$) then the machine goes from the state v to the state u by means of input i. Since β has been defined only for vertices in V, we extend here β so that for $u\in V_I$ $\beta(u)$ may be a certain "color" which is different from any other color allocated to the other vertex than u. We call such a sequential Moore machine the conjugate machine of A.

#2. Relationships between S_β and equivalence relation in the standard sequential Moore machine

In the text books of automata theory (for example, Salomaa 1969), the noninitial sequential Moore machine is defined as a 5-tuple (Q,X,Y,f,g) where Q is the state set, X and Y are input and output alphabet respectively, $f:\ Q\times X\to Q$ is the state transition function and $g:\ Q\to Y$ is the output function.

A pair of states $p,q\in Q$ is defined to be equivalent (and denoted by $p\equiv q$) if and only if $g(p)=g(q)$ and for every $w\in X^*$ $g(f(p,w))=g(f(q,w))$ where $g(f(p,\lambda))=\lambda$ (λ is the null word).
The relation $\equiv$ is clearly an equivalence relation.

The notion of equivalence $\equiv$ can naturally be extended to an incompletely specified sequential Moore machine. That is, for an element $a \in X$ if $f(p,a)$ is not defined, $f(q,a)$ should be also undefined in order to be the case where $p \equiv q$.

We describe here the definition of the structural equivalence relation in the colored network. Let $A=(G, \beta)$ be a colored network with $G=(V',E)$. An equivalence relation R on VxV is called a structural equivalence relation of A if and only if the following condition holds: if $(u,v) \in R$ then $\beta(u)=\beta(v)$ and $(E(u),E(v)) \in R$, where $E(u)$ and so on is the abbreviation of $E_1(u),E_2(u),\ldots,E_k(u)$ and so on.

Proposition 1

Let $A=(G, \beta)$ be a finite automata network and $\equiv(\beta)$ be the equivalence relation in the conjugate machine $\bar{A}$ of A. Then $\equiv(\beta) \in \mathfrak{S}_A$.

Proof

If $p \equiv (\beta) q$, then $\beta(p)=\beta(q)$ and $E(p) \equiv (\beta) E(q)$. The latter holds from the fact that in terms of sequential Moore machine, if $p \equiv q$ then for every $w \in X^*$ $f(p,w)=f(q,w)$. So from the definition of SER, $\equiv(\beta)$ is a SER.

Corollary 1

If A is structurally reduced, then its conjugate machine is reduced in the sense of the ordinary automata theory.

Proposition 2

For every R in $\mathfrak{S}_A$, if $(p,q) \in R$ then $p \equiv (\beta) q$. That is, any relation in $\mathfrak{S}_A$ is a refinement of $\equiv(\beta)$.

Proof

If $(p,q) \in R$ then $\beta(p)=\beta(q)$ and $(E(p),E(q)) \in R$. So $\beta(E(p))=\beta(E(q))$. By induction for every nonnull string w from E^*, $\beta(pw)=\beta(qw)$. This means $p \equiv (\beta)q$. Here pw denotes, as in the paper (1), $E_{i_h}(E_{i_{h-1}}(\ldots(E_{i_1}(p))\ldots)$ where $w=E_{i_1}E_{i_2}\ldots E_{i_h}$.

A colored network (G,β) is said to be structurally reduced (s-reduced) if and only if $\mathfrak{S}_\beta=\{R_0\}$. (In the case of element uniform network, A is s-reduced if $\mathfrak{S}_A=\{R_0,R_U\}$.)

A reduced Moore machine has no pair of equivalent states.

Corollary 2

If the conjugate machine of A is reduced, then A is s-reduced.

Proposition 3

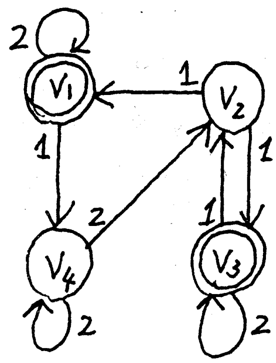
There is a case where $\mathfrak{S}_\beta$ contains other relations than $\equiv(\beta)$.

Proof

Look at the following example.

$V=\{v_1,v_2,v_3,v_4\}$

	E_1	E_2
v_1	v_2	v_1
v_2	v_3	v_4
v_3	v_2	v_3
v_4	v_1	v_4



$\beta(v_1)=\beta(v_3)=a$ $\quad \beta(v_2)=\beta(v_4)=b$

In this example $\equiv(\beta)=([v_1,v_3],[v_2,v_4])$, but $\mathfrak{S}_\beta$ also contains $([v_1,v_3],v_2,v_4)$.

Let us investigate here the important role of $\mathfrak{S}_G$, or the set of SER's with uniform elements allocated to a fixed graph G. When we write $A=(G, \beta)$, we assume that β is nonuniform or $|\mathrm{image}(\beta)| \geq 2$.

<u>Definition 3</u>

Take an arbitrary R from $\mathfrak{S}_G$. A coloring function β is said to be consistent with R if and only if $(p,q) \in R$ implies $\beta(p)=\beta(q)$. β is called a maximum coloring function consistent with R if and only if $(p,q) \in R$ implies $\beta(p)=\beta(q)$ and $(p,q) \notin R$ implies $\beta(p) \neq \beta(q)$.

<u>Proposition 4</u>

Let R be an arbitrary relation from $\mathfrak{S}_G$. If β is consistent with R then $R \in \mathfrak{S}_\beta$.

<u>Proposition 5</u>

For every β, $\mathfrak{S}_\beta \subsetneq \mathfrak{S}_G$.

<u>Proposition 6</u>

Let R be an arbitrary relation from $\mathfrak{S}_G (R \neq R_U)$. Then there exists a Moore machine whose state transition diagram is $\overline{G}$ and such that $R = \equiv(\beta)$ where β is the output function.

<u>Proof</u>

As such a β, take the maximum coloring function consistent with R.

Note: A Moore machine treated in Porposition 6 is called a Moore machine on G. By defining the output function in many ways, we have as many distinct Moore machine on G.

Proposition 7

Let G be a graph whose conjugate machine is reduced (the output function is defined in some way). Then $\mathfrak{S}_G$ is not necessarily $\{R_0, R_U\}$.

Since if $\mathfrak{S}_G = \{R_0, R_U\}$, then $\mathfrak{S}_\beta = \{R_0\}$ for every β, it is meaningful to investigate the properties of $\mathfrak{S}_G$.

#3 Structurally reduced colored networks

In the previous section we investigated the structure of $\mathfrak{S}_\beta$ of finite colored networks in contrast to the finite sequential Moore machine. Here we are going to treat some properties of structurally reduced (not necessarily finite) networks, in particular element uniform networks.

Property 1

If an element uniform network G=(V,E) is s-reduced, then any augmented network G'=(V,E') is also s-reduced, where $E' \supset E$. Note that the converse of this property does not holds.

Property 2

For every G=(V,E) there exists E' such that $E' \supset E$ and G'=(V,E') becomes s-reduced.

Property 3

When the number of vertices is prime, if there is a Hamiltonian circuit having the label i (i=1,2,...,or k) in the inverted graph of G, G is s-reduced.

Property 4

When k=1, only the rings with prime length can be s-reduced.

Property 5 (a corollary of Theorems 4.1 and 4.5 of (2))

Suppose that G is a group graph. Then G is s-reduced if and only if G has no nontrivial subgroup. Since every (countably) infinite group has at least one nontrivial subgroup, every infinite group graph is not s-reduced.

Property 6

A colored network $A=(G, \beta)$ is s-reduced if and only if there exists a mapping $\gamma : H \rightarrow M$ such that $\mathcal{B}_{\gamma \cdot \beta} = \{R_0, R_U\}$.

Property 7

An s-reduced graph is necessarily strongly connected in the sense of automata theory.

#4 Concluding remarks

1) As was discussed in Section 2, in the finite case our definition of structural equivalence relation was proved to be very similar to the notion of state equivalence in the Moore machine. What happens if we extend such a finite automata theoretic notion to the infinite system ?

2) Beside the theoretical work we developed a computer program for generating $\mathfrak{S}_G$ and are continuing computer experiments using it. In particular we want to get a conclusion about how "often" the s-reduced networks are generated. Though we have not made enough experiments, it seems that the ratio of the number of s-reduced networks to that of strongly connected graphs becomes near 1 when the number of vertices becomes large.

3) Though we have established an algorithm for generating $\mathcal{S}_{G_T}$ (3), which is not the same as that used at the computer experiment, we have not evaluated the efficiency of the algorithm. It is also a problem to be soved to obtain an efficient algorithm for determining whether a given colored network is s-reduced or not.

References

(1) T.Saito and H.Nishio, "Structural and Behavioral Equivalence Relation in Automata Networks" 1984 Winter LA Symposium, Kyoto

(2) T.Saito and H.Nishio, "Structural and Behavioral Equivalence Relation in Automata Networks--2nd Report" 1984 Summer LA Symposium, Hakone

(3) T.Saito and H.Nishio, "Structural and Behavioral Equivalence Relations in Automata Networks" (to appear)